Electronic Supplementary Materials

For https://doi.org/10.1631/jzus.A2200311

GPU-accelerated vector-form particle-element method for 3D elastoplastic contact of structures

Wei WANG[1,2], Yanfeng ZHENG[1,3,4]✉, Jingzhe TANG[1], Chao YANG[1,4], Yaozhi LUO[1,4]

[1]*College of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, China*

[2]*Center for Balance Architecture, Zhejiang University, Hangzhou 310028, China*

[3]*Architectural Design and Research Institute of Zhejiang University Co., Ltd., Hangzhou 310028, China*

[4]*Key Laboratory of Space Structures of Zhejiang Province, Hangzhou 310058, China*

✉ Yanfeng ZHENG, yanfeng39@zju.edu.cn

S1 J2 plasticity model

In addition to the elastic constitutive model, the J2 plasticity model with isotropic hardening is utilized in this paper. The yield criterion is

$$f(\boldsymbol{\sigma},\kappa)=\sqrt{2J_2}-\kappa\le 0, \tag{S1}$$

where J_2 denotes the second invariant of the deviatoric stress tensor, κ represents the plastic internal variable, which can be regarded as a linear function of the cumulative plastic strain:

$$\kappa=\kappa_0+\frac{2}{3}H^{\mathrm{iso}}\lambda,\quad \lambda=\int_t|\dot{\boldsymbol{\varepsilon}}^p|\,\mathrm{d}t, \tag{S2}$$

where $\dot{\boldsymbol{\varepsilon}}^p$ denotes the equivalent plastic strain rate, λ represents the cumulative plastic strain, $\kappa_0=\sqrt{\frac{2}{3}}\sigma_y$ denotes the initial plastic internal variable, σ_y represents the initial yield stress, and H^{iso} denotes the isotropic plastic modulus.

S2 Search for closest target particles

The algorithm for the parallel contact search procedures is summarized in Algorithm S1.

Algorithm S1 Search for closest target particles

// N_{co}: number of contactor particles
// i_{co}: index of contactor particle P_c
parallel for $(0\le i_{co}\le N_{co})$
    Obtain the index of cell that contains particle i_{co}: $(\hat{i}_{cx},\hat{i}_{cy},\hat{i}_{cz})$
    Assign a large value to the minimum distance $d_{\min}$: $d_{\min}=1\times10^{10}$

Initialize the closest target particle: $i_t^* = -1$

// Loop over target particles within cell i_{co} *and the* 26 *adjacent cells*

for $\hat{i}_{cz} - 1 \le i_{cz} \le \hat{i}_{cz} + 1$ **do**

for $\hat{i}_{cy} - 1 \le i_{cy} \le \hat{i}_{cy} + 1$ **do**

for $\hat{i}_{cx} - 1 \le i_{cx} \le \hat{i}_{cx} + 1$ **do**

Compute cell index i_c of the current cell

Read the first particle stored in cell i_c: $i_t = pHead[i_c]$

while i_t is a target particle **do**

Compute distance between the two particles: $d = \mathrm{dist}(i_{co}, i_t)$

if d < $d_{\min}$ **do**

$d_{\min} = d, \quad i_t^* = i_t$

end if

Read the next contact particle: $i_t = pNext[i_t]$

end while

end for

end for

end for

end parallel for

---

*The keyword pair "**parallel for** ... **end parallel**" indicates that the codes within it are executed in parallel in GPU threads.

S3 Normal and tangential penalty stiffness

The normal and tangential stiffness can be evaluated using the following equations (Hallquist et al., 1985):

$$k_{n,i} = s_n \frac{K_i A_i^2}{V_i}, \tag{S3}$$

$$k_{t,i} = s_t \frac{K_i A_i^2}{V_i}, \tag{S4}$$

where V_i is the volume of the hexahedral element containing target face S_i, K_i and A_i denote the bulk modulus and area of face S_i, respectively, s_n and s_t denote the scaling factors for the normal and tangential penalty stiffness, respectively. The default values of s_n and s_t are 1.0.

S4 Flowchart of parallel FPM solvers

The flowchart of the parallel FPM solvers is shown in Fig. S1. The computational procedures in the solvers are performed in GPU threads, and the thread counts are listed in Table S1.

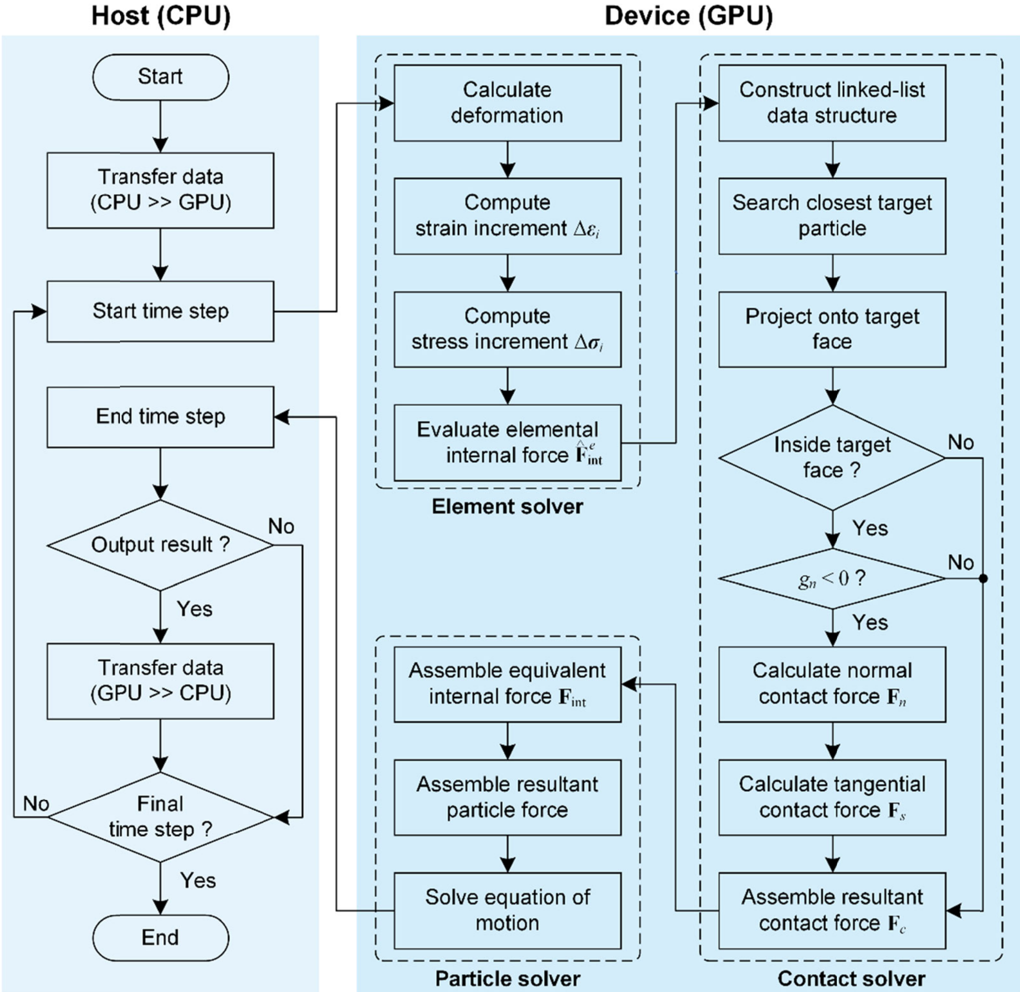


Fig. S1 Flowchart of the parallel FPM solvers

Table S1 Number of threads for each computational procedure

Solver	Number	Procedure	Number of threads
Element	1	Calculate deformation of the elements	N_e
	2	Evaluate strain increments $\Delta\boldsymbol{\varepsilon}$	$N_e \times N_{\text{int}}$
	3	Evaluate stress increments $\Delta\boldsymbol{\sigma}$	$N_e \times N_{\text{int}}$
	4	Obtain elemental internal forces $\hat{\mathbf{F}}^e_{\text{int}}$	N_e
Contact	1	Construct connected list data structure	N_{cp}
	2	Search for the closest target particles for contactor particles	N_{co}
	3	Project contactor particles onto target faces	$N_{co} \times S_{\max}$

	4	Calculate normal contact forces	$N_{co} \times S_{\max}$
	5	Calculate tangential contact forces	$N_{co} \times S_{\max}$
	6	Assemble resultant contact forces	N_{cp}
	1	Assemble equivalent internal forces $\mathbf{F}_{\text{int}}$	N_p
Particle	2	Assemble resultant particle forces	N_p
	3	Solve motion equations	N_p

N_e : element count

N_p : particle count

N_{int} : integration point count

N_{cp} : contact particle count (including contactor particles and target particles)

N_{co} : contactor particle count

$S_{\max}$: maximum face count linked by each target particles

S5 Verification example: elastic contact

The frictional contact between two elastic beams is considered in this example, as shown in Fig. S2, which has been studied in the literature (Litewka and Wriggers, 2002). Beam 2 is initially located above the center of cantilever beam 1. The length of both beams is 1000 mm. The section sizes of beam 1 and beam 2 are 100 mm × 100 mm and 50 mm × 50 mm, respectively. These two beams are perpendicular to each other, and the initial gap between them is 5.0 mm. The horizontal displacement (Δx = 400 mm) and vertical displacement (Δz = -200 mm) are applied to both ends of beam 2 from t = 0 to t = 1 s, which makes beam 2 move toward the free end of beam 1.

The elastic material properties of the beams are given by Young's modulus 300 MPa and Poisson's ratio 0.17. Beam 1 and beam 2 are discretized into 2,160 and 1,280 hexahedral elements, respectively. A frictionless case and two frictional cases (μ = 0.5, 1.0) are investigated. Both scaling factors s_n and s_t are set to 1.0. The mass damping coefficient is set to 1×10^4.

The deformed configurations with contour plots of vertical displacement at 0.3 s, 0.6 s, and 1.0 s are shown in Fig. S3. It can be found that the deformation of beam 2 is greatly affected by the friction coefficient. The horizontal displacement history of point A (the center point of beam 2, see Fig. S2) is presented in Fig. S4 for different friction coefficients. It can be observed that the horizontal displacement of point A decreases with the increase of friction coefficient, which can also be concluded from Fig. S3. The displacement curves obtained by FPM match well with that obtained by FEM (Litewka and Wriggers, 2002). Thus, the proposed method is effective in modeling elastic frictional contact of structures.

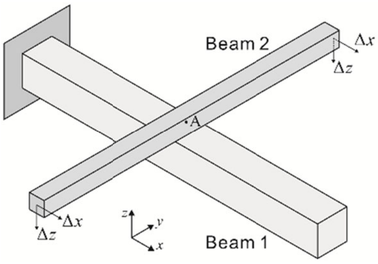
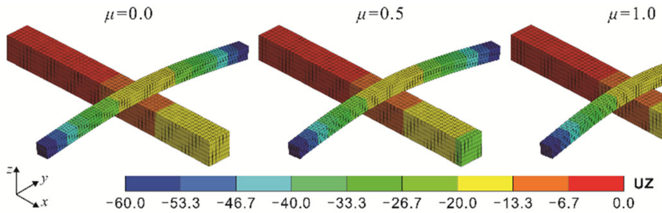
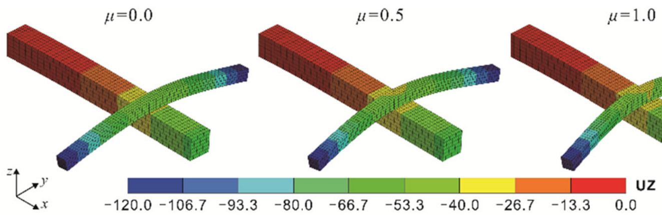


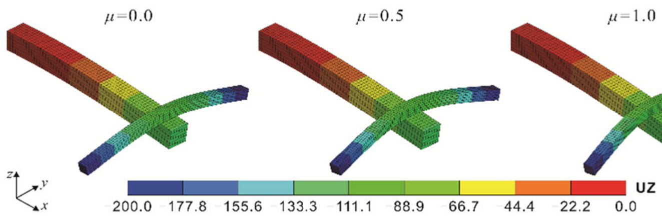
Fig. S2 Contact between two beams: geometry



(a) $t = 0.3$ s



(b) $t = 0.6$ s



(c) $t = 1.0$ s

Fig. S3 Contact between two beams: contour plots of vertical displacement (unit: mm)

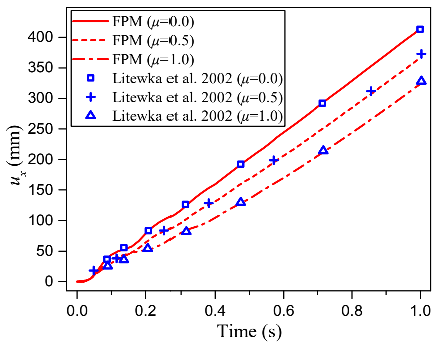


Fig. S4 Contact between two beams: horizontal displacement history of point A

S6 Computer configuration

The efficiency tests are performed on a computer with double precision. The configuration of the computer is listed in Table S2.

Table S2 Computer configuration

Name	Parameter
OS	Windows 10 64-bit
CPU	Intel® Core™ i7-4790K @ 4.00 GHz
RAM	Kingston 16 GB DDR3
GPU	NVIDIA Titan V (5,120 CUDA cores, 12 GB memory)

S7 Efficiency test: large-scale quasi-static elastic contact

This example is adapted from the two-layer pinched cylinder example (Puso, 2004). As shown in Fig. S5, two opposing concentrated forces F = 400 N are applied to the two opposing center points of a multilayer cylinder. The average radius, thickness, and length of the cylinder are 300 mm, 6 mm, and 600 mm, respectively. The elastic material properties are given by Young's modulus 30 MPa and Poisson's ratio 0.3. The mass damping coefficient is set to 20.0, and the scaling factor sn is set to 0.1. The friction between contact surfaces is ignored. Only one eighth of the cylinder is modeled due to symmetry.

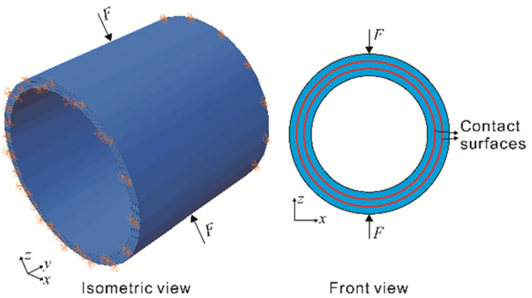


Fig. S5 Geometry of the multilayer pinched cylinder

A two-layer cylinder is studied first to demonstrate the results. Each layer has a thickness of 3 mm, and is discretized into 1,350 (45×30×1) hexahedral elements. The displacement contour is shown in Fig. S6. It can be found that the displacement field is continuous at the contact surfaces of the two-layer cylinder as expected. The relation between the dis-placement of the upper corner node and the concentrated force F is presented in Fig. S7, indicating that the load-displacement curves obtained by FPM and Abaqus agree well.

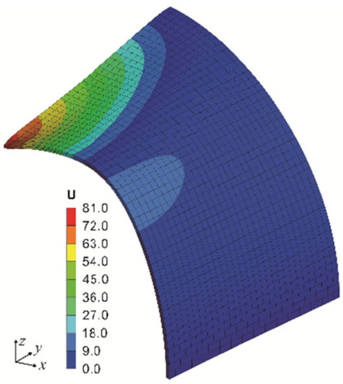


Fig. S6 Two-layer pinched cylinder: contour plot of displacement (unit: mm)

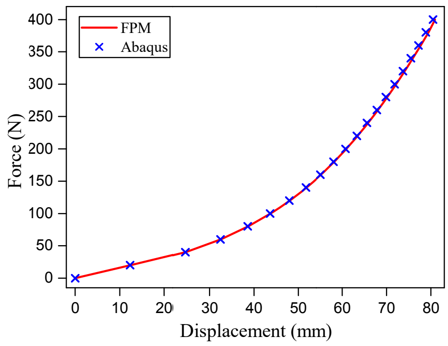


Fig. S7 Two-layer pinched cylinder: load-displacement curve

To test the efficiency of the proposed method in quasi-static elastic contacts, multilayer pinched cylinders with same layer thickness (0.6 mm) but different layer counts are investigated. Each layer is discretized into 135,000 (450×300×1) hexahedral elements. All cases are analyzed with a total number of 1,000 time steps. The computational times of contact calculation and overall computation are given in Table S3. Specifically, the CPU time consumed by Abaqus contact solver, denoted as T_c, is approximated by the following equation instead of directly obtained in Abaqus

$$T_c = T_{\text{involve\_contact}} - T_{\text{no\_contact}}, \tag{S5}$$

where $T_{\text{involve\_contact}}$ and $T_{\text{no\_contact}}$ denote the computational times considering and not considering contact, respectively.

Table S3 Multilayer pinched cylinder: computational times in seconds

Number of layers	Number of elements	Abaqus (CPU)		FPM (CPU)		FPM (GPU)	
		Contact	Total	Contact	Total	Contact	Total
2	270,000	208.5	301.0	72.1	738.4	0.9	6.2
4	540,000	641.0	826.0	221.0	1584.7	2.1	12.0
6	810,000	1090.5	1366.0	376.6	2466.6	3.3	18.0
8	1,080,000	1528.0	1895.0	502.4	3286.9	4.5	23.5

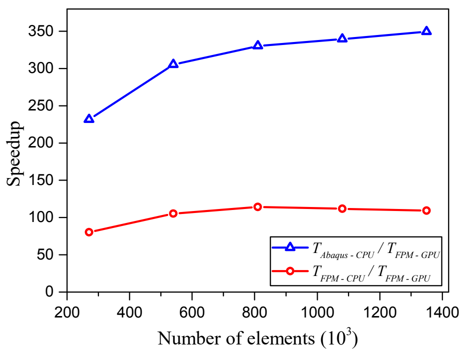
10	1,350,000	1991.0	2453.0	623.2	3806.1	5.7	29.6

The speedups of the contact calculation are shown in Fig. S8a. With the increase of element count, the speedup of the parallel FPM over the serial FPM grows at first and reaches the maximum value of 113 when the number of elements exceeds 0.8 million, demonstrating that the GPU parallel implementation significantly accelerates the parallel contact solver. Meanwhile, the maximum speedup of the parallel FPM over Abaqus is 346 when the number of elements is approximately 1.3 million. It indicates that the serial FPM contact solver is faster than the contact solver in Abaqus, which might be mainly owing to the bucket sort approach adopted in the contact search procedures.

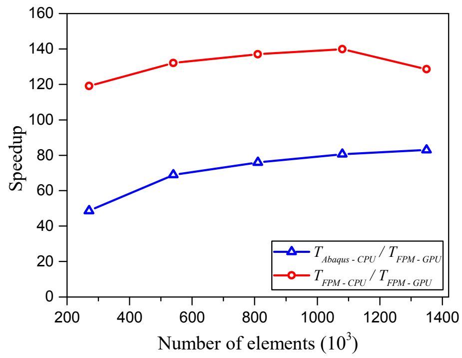
The speedups of the overall computation are presented in Fig. S8b. The maximum speedups of the parallel FPM over the serial FPM and Abaqus are 140 and 82, respectively. Thus, the parallel FPM is proved to be efficient. It can also be found that the serial FPM solver is less efficient than the Abaqus/Explicit solver. This might be because FPM needs more computations on the fictitious reverse motion compared with the method utilized in Abaqus.

Fig. S8 also indicates that the GPU is running at full load when the number of elements reaches approximately 0.8 ~ 1.4 million, and the improvement of computational efficiency gradually stabilizes. Similar observations can be found in the literature (Dong et al., 2015). One can achieve higher speedups by using GPUs with larger device memory and more CUDA cores.

Fig. S9 depicts the proportions of contact calculation time in the overall computational time. As the element count increases, the three time-consumption percentages in Fig. S9 increase a little bit at first and nearly remain unchanged after-wards. The contact calculation accounts for ap-proximately 80%, 16%, and 19% in Abaqus, serial FPM, and parallel FPM, respectively. Therefore, the proposed parallel implementation dramatically reduces the proportion of contact calculation in overall computation compared with commercial finite element codes.



(a) speedup of contact calculation



(b) speedup of overall computation

Fig. S8 Multilayer pinched cylinder: speedups

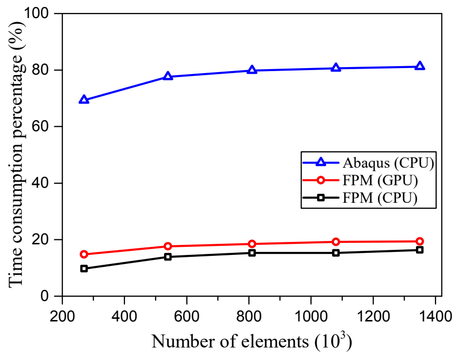


Fig. S9 Multilayer pinched cylinder: proportion of contact calculation time

S8 Efficiency test: large-scale dynamic elastoplastic contact

The computational times for the efficiency test of the dynamic elastoplastic contact are given in Table S4.

Table S4 Multilayer plate under impact loading: computational times in seconds

Number of layers	Number of elements	Abaqus (CPU)		FPM (CPU)		FPM (GPU)	
		Contact	Total	Contact	Total	Contact	Total
4	250,000	266.0	368.0	123.6	706.6	1.1	6.2
8	500,000	591.0	794.0	280.4	1438.0	2.1	11.6
12	750,000	921.0	1224.0	437.8	2221.8	3.0	16.9
16	1,000,000	1257.0	1657.0	594.0	2904.0	4.0	22.1
20	1,250,000	1590.0	2102.0	751.2	3664.0	5.1	27.7

References

Dong Y, Wang D, Randolph MF, 2015. A gpu parallel computing strategy for the material point method. *Computers and Geotechnics*, 66:31-38. https://doi.org/10.1016/j.compgeo.2015.01.009

Hallquist J, Goudreau G, Benson D, 1985. Sliding interfaces with contact-impact in large-scale lagrangian computations. *Computer Methods in Applied Mechanics and Engineering*, 51(1-3):107-137. https://doi.org/10.1016/0045-7825(85)90030-1

Litewka P, Wriggers P, 2002. Frictional contact between 3d beams. *Computational Mechanics*, 28(1):26-39. https://doi.org/10.1007/s004660100266

Puso MA, 2004. A 3d mortar method for solid mechanics. *International Journal for Numerical Methods in Engineering*, 59(3):315-336. https://doi.org/10.1002/nme.865